以二元搜尋法為基礎的線性區塊碼解碼

Based on Binary Search Decoding Scheme for Linear Block Codes

洪偉文　林義楠　黃植振

Wei-Wen Hung, Yi-Nan Lin, J.-J. Huang

摘要

傳統上在通道上錯誤更正用的線性區塊碼其解碼演算法，主要是利用解聯立方程式來求出正確碼字。因訊息經過傳輸通道後，某些訊息受雜訊干擾，接收端會判斷出錯誤的徵狀，而這些演算法乃解出其錯誤位元的位置，依其解出的錯誤位置，加以更正還原成原來傳送的訊息。本文建構在最大相似度的解碼上以二元搜尋法為基礎的解碼機制，其複雜度只需 O(k)，其中 k 為區塊訊息長度，為一即時的解碼機制。比較完全搜尋法需 $O(2^k)$及傳統演算法需反覆疊代式地求解方程式；本文所提的方法，除可節省計算電路的成本外亦可達到即時解碼的效果，但需要有 2^k 個標準碼字的儲存空間，因此適合用在短碼字線性區塊碼的錯誤保護的系統上。

關鍵詞：二元樹搜尋法、二元決策樹、最大相似度解碼、短線性區塊碼

ABSTRACT

The channel control coding algorithms can be correcting error bits which are corrupted by the channel noises by solving joint equations and then find out the error bit positions. Finally, the mechanism can get an original transmitted source message by summation of received message and error pattern in the last stage. A new method based on maximal likelihood decoding is proposed in which the binary search is used to develop a decoding scheme based on a binary decision tree. This decoding mechanism can determine the received codeword which is correct or not with time complexity about approximate O(k) which k is message symbol length, it is a real time decoding scheme. Comparing with the full search method which needs $O(2^k)$ and traditional algorithms need iterate to solve the joint equations which have hard computing circuit. The proposed method is much more efficient, but it needs $O(2^k)$ device to store the standard codeword, it is suitable in the short codeword of linear block codes.

Keywords： binary search tree, binary decision tree, maximal likelihood decoding, short linear block code

1 簡介

錯誤更正碼主要用來保護數位訊息在傳輸過程中，受到雜訊干擾下，能適時地予以更正其錯誤位元的一種機制[1]。錯誤更正碼是在傳送的訊息中，加入了查核位元；即為編碼，而這些查核的位元則可以在接收端解碼時，發揮其錯誤偵測及錯誤更正的效果。由提高可靠性的角度來看，錯誤控制更正碼是衛星通訊、行動通訊、衛星廣播、文字廣播領域不可缺少的技術，對於數位資料儲存系統而言，採用錯誤更正保護的機制對儲存的資料加以防護，可進一步提高資料的可靠度。目前，錯誤控制更正碼的技術已成功地應用在藍芽無線通訊系統[2]、個人無線通訊系統之 AMPS、NA-TDMA、CDMA、GSM [3]及 CD 資料儲存系統[4]等數位傳輸的技術方面。

洪偉文　明志科技大學電子工程系教授
林義楠　明志科技大學電子工程系講師
黃植振　明志科技大學電子工程系助理教授

以下將介紹本文相關的章節內容，第二節線性區塊碼編碼與其徵狀解碼方式，第三節介紹二元搜尋法基本原理，及建構一二元決策樹為基礎的解碼演算法。第四節以(6,3)的線性區塊碼來實現本文的解碼機制，最後做一結論探討。

2 線性區塊碼

線性區塊碼主要是將欲傳送的訊息加入一些冗餘的查核位元。這些查核位元的目的除了可以使得編出來的碼字均勻地分佈在整個碼字的空間外，更可以用來校正收到的碼字是否為一正確合法的碼字,以達到錯誤偵測或錯誤更正的效果。

2.1 編碼機制

在線性區塊碼中，有一些比較著名的碼。如：BCH 碼、RS 碼及 RM 碼[1]。今考慮具單一錯誤訂正能力的(6, 3)BCH 碼，其碼字長度為 6 個位元，其中訊息長度為 3 個位元，因此其查核位元需要 3 個，且此編碼具有最小漢明距離 d_{min} 為 3。由錯誤更正能力 $t = \left\lfloor \frac{d_{min} - 1}{2} \right\rfloor$ 及錯誤偵測能力 $e = d_{min} - 1$。因此，此編碼具有偵錯 2 個位元錯誤或是更正 1 個位元錯誤的能力。由(n, k)線性方塊碼的產生原理，若原始訊息向量為 **M**，其維度為 1 × k，即有 k 個位元的訊息量要進行編碼，其與一個 k × n 的生成矩陣 **G**，透過矩陣乘積的運算，可得到一長度為 n 個位元的碼向量 **X**，亦即 **X** = **M**·**G**。又若生成矩陣 **G** 具有[**P** | $\mathbf{I_K}$]的結構時，其中 $\mathbf{I_K}$ 是 k×k 的單位矩陣，而 **P** 是一個線性獨立的 k × (n - k)之子矩陣，則可運算出一個具有前 k 個位元為輸入編碼器的訊息位元串，及後半段為 n - k 個查核位元串分開的系統區塊碼(systematic block code)，即 **X** = [$m_1, m_2, \ldots, m_k, c_1, c_2, \ldots, c_{n-k}$]。其中 m_i，$1 \le i \le k$ 為 k 個位元的訊息；c_j，$1 \le j \le n-k$ 為 n-k 個查核位元。針對此生成矩陣 **G** 可產生一系統方塊碼 **X**。

若 $\mathbf{G} = \begin{bmatrix} 0\,1\,1\,1\,0\,0 \\ 1\,0\,1\,0\,1\,0 \\ 1\,1\,0\,0\,0\,1 \end{bmatrix}_{3\times 6} = [\mathbf{P} \mid \mathbf{I_3}]$

表 1：(6, 3)線性區塊系統碼：**X** = **M**·**G**

M	**X**	w(**X**)	$(\mathbf{X})_{10}$
000	000000	0	0
100	011100	3	28
010	101010	3	42
110	110110	4	54
001	110001	3	49
101	101101	4	45
011	011011	4	27
111	000111	4	7

其中，w(**X**)：碼字的權重，即碼字的漢明距離；$(\mathbf{X})_{10}$：碼字的十進制值。

2.2 徵狀(syndrome)解碼

當發送端傳送某一個碼字 **X**，在二位元對稱通道傳輸模式下，接收端收到向量經硬決策後為 **Y** 時，若 **Y** ≠ **X**，則表示傳輸過程中，因雜訊的干擾而造成位元錯誤發生。

對於任一(n, k)線性系統方塊碼，其生成矩陣為 **G** = [**P** | $\mathbf{I_k}$]$_{k\times n}$ 時，則存在一個(n - k)×n 的正交對應的同位檢驗矩陣 **H** = [$\mathbf{I_{(n-k)\times n}}$ | $\mathbf{P^T}$]，此同位檢查矩陣的特性為:若 **Y** 為一合法的碼字時，則 $\mathbf{YH^T} = [0\ 0\ \ldots\ 0]_{1\times(n-k)}$；反之，若 **Y** 不為一合法的碼字時，則 $\mathbf{YH^T} \neq [0\ 0\ \ldots\ 0]_{1\times(n-k)}$。故在解碼的過程中，接收端一收到向量 **Y**，則先計算 **S** = $\mathbf{YH^T}$，此 **S** 稱為 **Y** 的徵狀。若 **S** ≠ **0** 時，則表示傳輸的過程中，有錯誤的發生。反之 **S** = **0**，則 **Y** = **X**，即表示傳輸的過程中沒有錯誤發生；或者 **Y** 可能解碼為另一個合法的碼字而形成了無法偵測的錯誤。

根據錯誤位元的最大發生機率，則可找出錯誤型態向量 **E** 與徵狀向量 **S** 間之一對一的關係。故在計算出徵狀向量 **S** 後，即可進一步地計算出錯誤發生的位置 **σ**，再將此錯誤型態的向量 **E** 與接收的向量 **Y** 相加，即可解出一合法的碼字。其解碼系統架構圖[6]，如圖 1 所示。

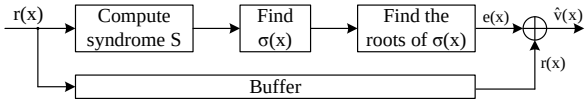

圖 1. 線性區塊碼解碼系統架構圖

其中，**S** 表示對應於接收向量 **r(x)**的徵狀，**σ (x)** 為錯誤位置的多項式，**e(x)**為一錯誤多項式，及解碼後的碼字 $\hat{\mathbf{v}}(\mathbf{x})$ 。

3 以二元搜尋法為基礎的線性區塊碼之解碼機制

3.1 二元搜尋法(Binary Search) [5]

將所有合法的碼字，依其對應十進制值的大小，做遞增式的排列。而每次做鍵值的比對搜尋時，利用二分法把陣列分成兩半，然後由其中的一半去找尋。亦即，在一開始令 *low* 為陣列最小的索引值，而 *high* 為陣列中最大的索引，而其搜尋鍵值的索引位置為 $mid = [(low + high)/2]$。當搜尋比較發生鍵值 $k = key(mid)$ 時，則表示搜尋成功。若 $k < key(mid)$ 時，則需遞迴到陣列的前半部；亦即往索引 $low \sim high(= mid - 1)$ 的範圍繼續尋找。若 $k > key(mid)$ 時，則需遞迴到陣列的後半部；亦即往索引 $low(= mid + 1) \sim high$ 的範圍繼續尋找，每一次在陣列中找尋的次數為 $high - low + 1$。此外，剩下候選搜尋鍵值個數為 $\le (high - low + 1)/2$。即在剛開始候選鍵的個數有 *n* 個，在第一次遞迴呼叫後最多會有 *n/2* 個；在第二次遞迴呼叫後，它最多只會有 *n/4* 個；以此類推。因此，若 *T(n)*表示執行二元搜尋法所需的時間複雜度，則 *T(n)*可利用遞迴式表示如下：

$$T(n) = \begin{cases} b & ,if\ n < 2 \\ T(\frac{n}{2}) + b & ,n \ge 2 \end{cases} \qquad (1)$$

其中，*b* 為比較鍵值的常數時間，(1)式利用遞迴函數求解，可解出 $T(n) = b + b\log(n)$。即表示，二元搜尋演算法執行所需之時間複雜度為 $O(\log n)$。故今若有 $10^6 \approx 2^{20}$ 個合法鍵值，則最多花費的時間約為 $\log_2(2^{20})$，即 20 次的比較可完成鍵值的搜尋。

3.2 建立以二元搜尋法為基礎之解碼

以(6,3)的線性區塊為例，其真正合法的碼字只有 8 個(表 1)，即在可能的 2^6 為 64 個 6 位元的傳輸向量中，只有 8 個才是真正合法的碼字。在 k = 3 個位元的訊息，經由生成矩陣 **G** 的運算，可得到表 1，n = 6 個位元的碼字向量。以本文提出二元搜尋法為基礎的最大相似度解碼方式為例，可建立出一棵(6,3)線性區塊碼二元決策樹，如下圖 1 所示。

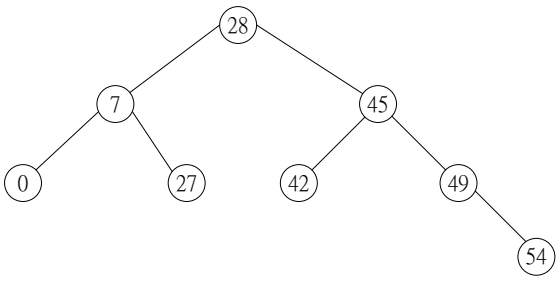

圖 2. (6, 3)線性區塊碼二元決策樹

3.3 以二元決策樹為基礎之最大相似度解碼演算法

解碼機制如下：

Step1.當接收機收到的碼字訊息 **R**，經由硬決策，即令 Ri ≥ 0 時，則 Yi = 1；否則 Yi = 0，可得到一個二元的碼字向量 **Y**。

Step2.將此二元的碼字向量 **Y**，轉換成對應的十進制鍵值 **V**。

Step3.搜尋此鍵值 **V**，是否存在於此二元決策樹中。一開始，由樹根節點開始進行比較，當鍵值 **V** 不等於拜訪到的節點值 **Xi** 時，則計算二個節點值的漢明距離 d = **(Y, Xi)**，當 d ≤ 1 時，即可立刻解碼出碼字為 **Xi**，解碼完成，離開。否則當 d > 1 時，則需比較 **V** 與 **Xi** 的大小，當 **V** 大，則往右分支走；若 **V** 小，則往左分支拜訪。一直遞迴拜訪下去，即重複步驟 3，直到拜訪到樹葉節點，仍未解出最佳的 **Xi** 時，即表示錯誤發生，離開。

例 1：當接收到 **R** = [0.1 -0.2 0.3 -0.5 0.6 -0.4]的訊號，其解碼流程如下：

(1). **Y** = [1 0 1 0 1 0]

(2). V = 42

(3). V ≠ 28，且 d = 4 > 1，走右分支。

(4). V ≠ 45，d = 3 > 1，走左分支。

(5). V = 42，則解碼完成，為一合法的碼字；且其碼字為[1 0 1 0 1 0]，因此原訊息為(0 1 0)。此解碼共花費 3 次的比較時間。

例 2：若接收到 **R** = [0.1 0.2 0.3 0.5 0.6 0.4]的訊號

大小時，其解碼如下：

(1) **Y** = [1 1 1 1 1 1]

(2) V = 63

(3). V ≠ 28，且 d = 3 > 1，走右分支。

(4). V ≠ 45，d = 2 > 1，走右分支。

(5). V ≠ 49，d = 3 > 1，走右分支。

(6). V ≠ 54，d = 2 > 1，因 54 為樹葉節點，故此碼字之解碼共花費了 4 次的比較，即此樹的最大高度，且解出的碼字為一不合法的碼字；即偵測出錯誤。

4 模擬結果

本節以(6,3)線性區塊碼為例，即一個區塊碼原訊息有 3 個位元，編碼後長度為 6 個位元，且 d_{min} = 3，即此碼具有一個位元的錯誤更正能力。透過實驗證實所提出方法的有效性；傳送端發送 10^6 個位元時，經(6,3)線性區塊碼編碼後，加以 BPSK(Binary Phase Shift Keying, 二準位相移鍵控調變)調變傳送，訊息在 AWGN(Additive white Gaussian noise, 白色高斯雜訊)的通道下傳遞，接收端接收訊息後將其解碼並與未加任何通道編碼的解碼效果相比，可得到錯誤位元發生率(BER)對訊號雜訊比的解碼增益圖，如圖 3 所示。由此證實，加入此錯誤控制編碼的機制 BCH(6, 3)，可以使得 BER 在 10^{-4} 時，其 Eb/No 約需 7dB，而未加入此通道編碼(Uncoded)，其解碼增益則為 9dB，因此加入此糾錯碼即可得到 2dB 以上的解碼增益。

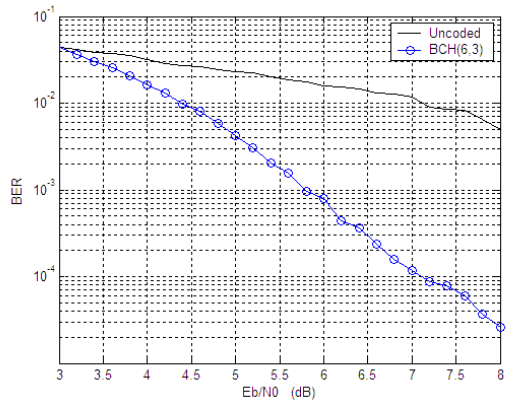

圖 2 BCH(6,3)解碼增益圖

5 結論

傳統使用徵狀解碼機制時，須先有一計算徵狀的電路；然後，利用解出的徵狀，來解出其對應的錯誤位置，這是一個聯立方程式的反覆求解。因此，在電路的設計與實現上有其困難的複雜度。本文提出以二元搜尋法為基礎的最大相似度線性區塊碼解碼方式，其解碼架構無需計算碼字徵狀及解方程式等複雜性的電路，只需透過碼字的比較，即可完成最大相似度的解碼，且其解碼的時間複雜度為 O(k)的比較運算，其中 k 為原始訊息位元數，除了可節省計算電路的成本外，亦可達到即時解碼的效果。此解碼的方式相當適合於具有行動計算能力的裝置上實現。但此演算的機制需有 2^k 個合法碼字的儲存空間，故適合於用在短碼字線性區塊碼的解碼。

6 參考文獻

[1] S. Lin and D. J. Costello Jr., “Error Control Coding: Fundamentals and Applications,” NJ: Prentice Hall, 1983.

[2] “Specification of the Bluetooth System,” v1.0 A, July 1999.

[3] D. J. Goodman, “Wireless Personal Communications Systems,” Addison-Wesley Longman, 1997.

[4] K. A. S. Immink, “Coding Techniques for Digital Recoders,” UK: Prentice Hall, 1991.

[5] R. F. Gilberg, B. A. Forouzan., “Data Structures: A Pseudo Code Approach with C++,” Brooks/Cole, 2001

[6] Robert H. Morelos-Zaragoza, “The Art of Error Correcting Coding,” Wiley, 2002